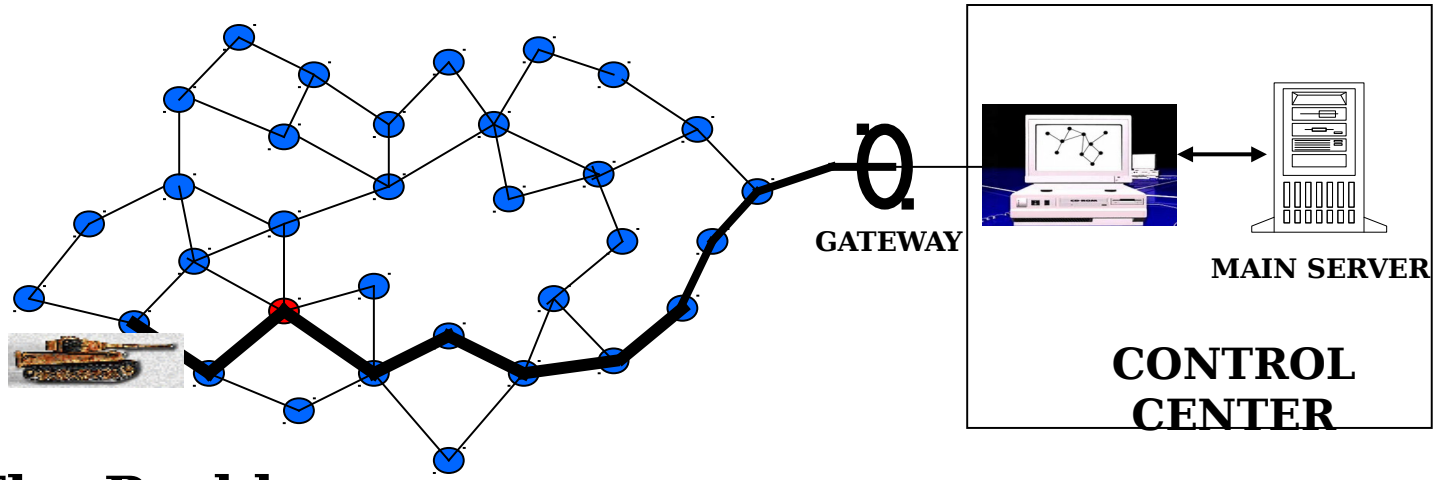
Coverage Algorithms

Mani Srivastava & Miodrag Potkonjak, UCLA
[Project: Sensor Information Science (RSC)]

&

Mark Jones, Virginia Tech
[Project: Sensor Information Technology (ISI-East)]

Sensor Network Coverage



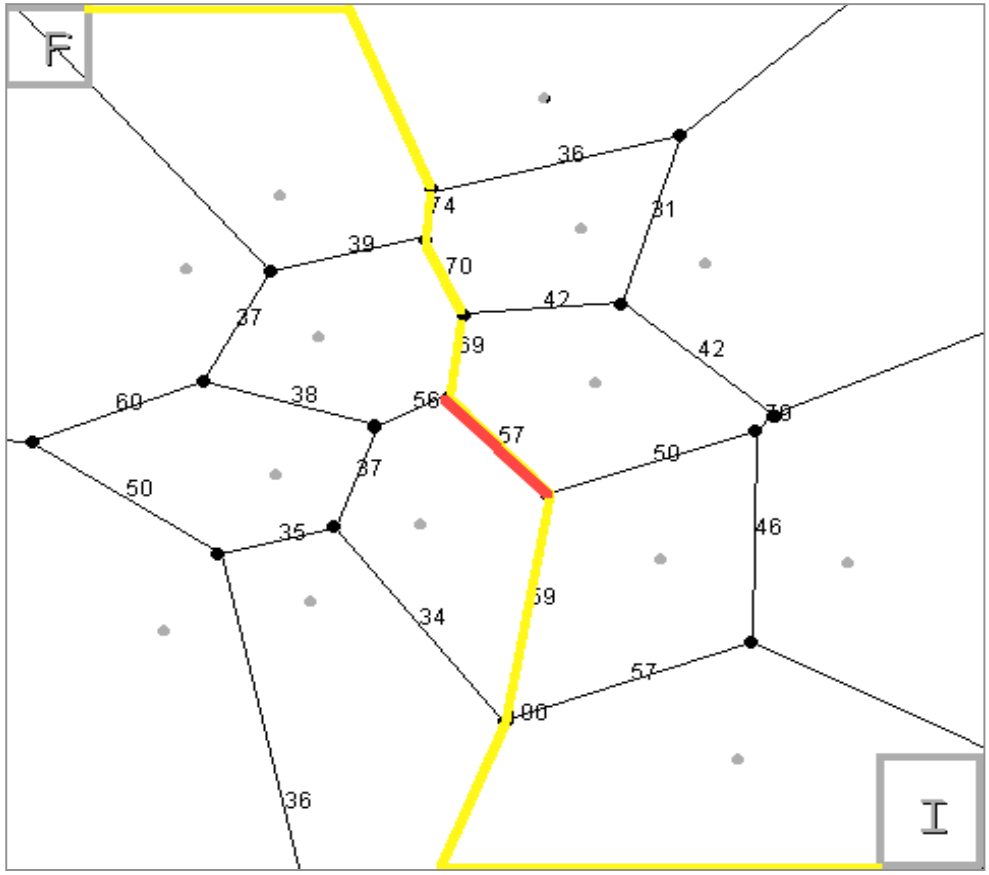
- **The Problem:**
  - **Given:**
    - **Ad hoc sensor field with some number of nodes with known location**
    - **Start and end positions of an agent**
  - **Want:**
    - **How well can the field be observed?**
- **Example usage**
  - **Commander**
    - **Weakest path: what path is the enemy likely to take?**
  - **Network manager**
    - **Weakest path: where to deploy additional nodes for optimum coverage?**
  - **Soldier in the battlefield**
    - **Strongest path: what path to take for maximum coverage by my command?**
    - **Weakest path: how to walk through enemy sensor net or through minefield?**

Summary of Our Work

- **Phase 1: distance to closest sensor [status: done, demonstrated]**
  - **Worst case coverage: *Maximal Breach Path***
  - **Best case coverage: *Maximal Support Path***
- **Phase 2: exposure to sensors [status: done, demonstrated]**
  - **Consider speed and distance**
  - **Worst case coverage: *Minimal Exposure Path***
- **Phase 3: localized distributed algorithms [status: current, experimented]**
  - **Query from user roaming in the sensor field**
  - **Computation done by the nodes themselves**
  - **Only relevant sensor nodes involved in the computation**
- **Phase 4 [future]**
  - **Probability of detection and its relationship with density**
  - **Heterogeneous sensors**
  - **Terrain-specific measured or statistical exposure models**

Closest Sensor Model: Maximal Breach Path

- **Problem: find the path between I & F with the property that for any point p on the path the distance to the closest sensor is maximized**
- **Observation: maximal breach path lies on the Voronoi Diagram Lines**
  - **by construction each line segment maximizes the distance from the nearest point**



Given*: Voronoi diagram *D* with vertex set *V* and line segment set *L* and sensors *S

Construct graph G(N,E):

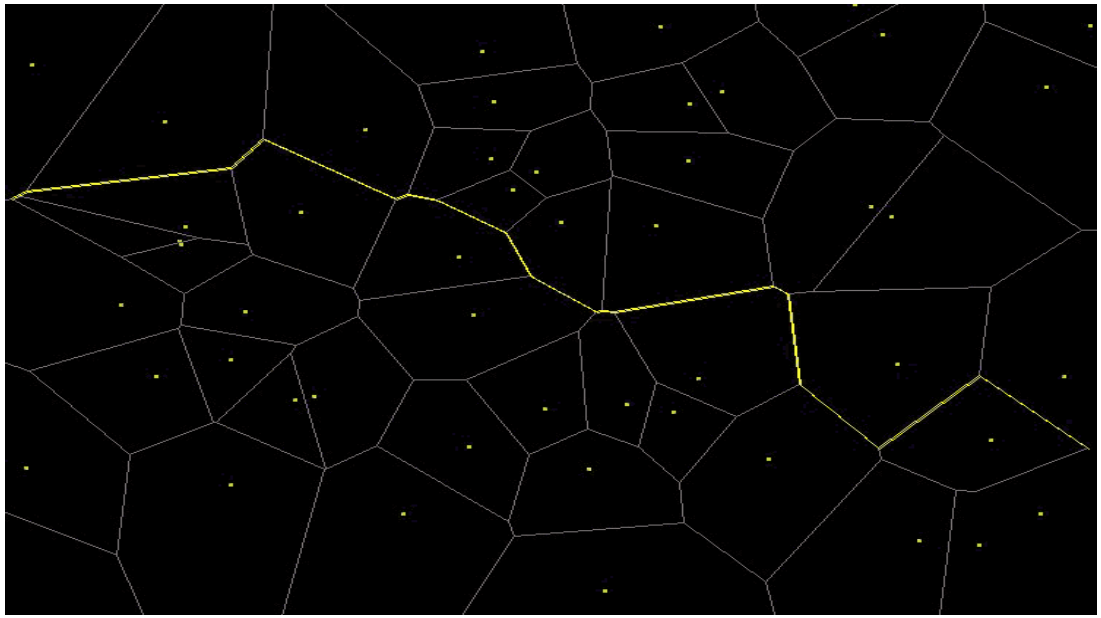
- **Each vertex $v_i \in V$ corresponds to a node $n_i \in N$**
- **Each line segment $l_i \in L$ corresponds to an edge $e_i \in E$**
- **Each edge $e_i \in E$, Weight(e_i) = Distance of l_i from closest sensor $s_k \in S$**

***Search for P_B*:**

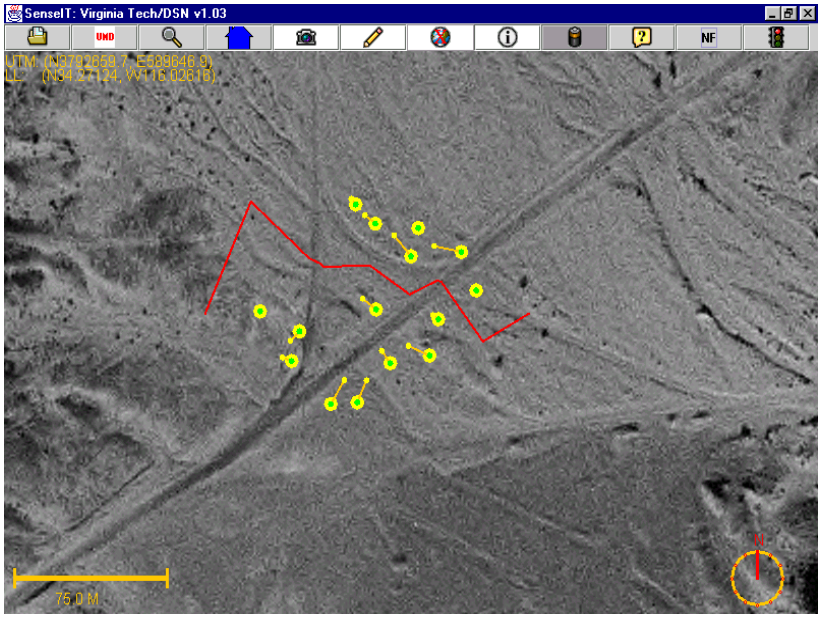
Check for existence of I→F path

Status

- **Simulation**
  - Demonstrated to Dr. Frank Fernandez in Spring 2000
- **Implementation**
  - Centralized coverage server
  - Integrated with the SensIT GUI (V. Tech.)
    - GUI passes node location
    - Server reports back the desired path
    - GUI displays sensor field coverage and breach paths
    - GUI also displays other status (e.g. battery) and controls nodes (e.g. activate)
  - Part of the SITEX demonstration in Summer 2000 & Spring 2001



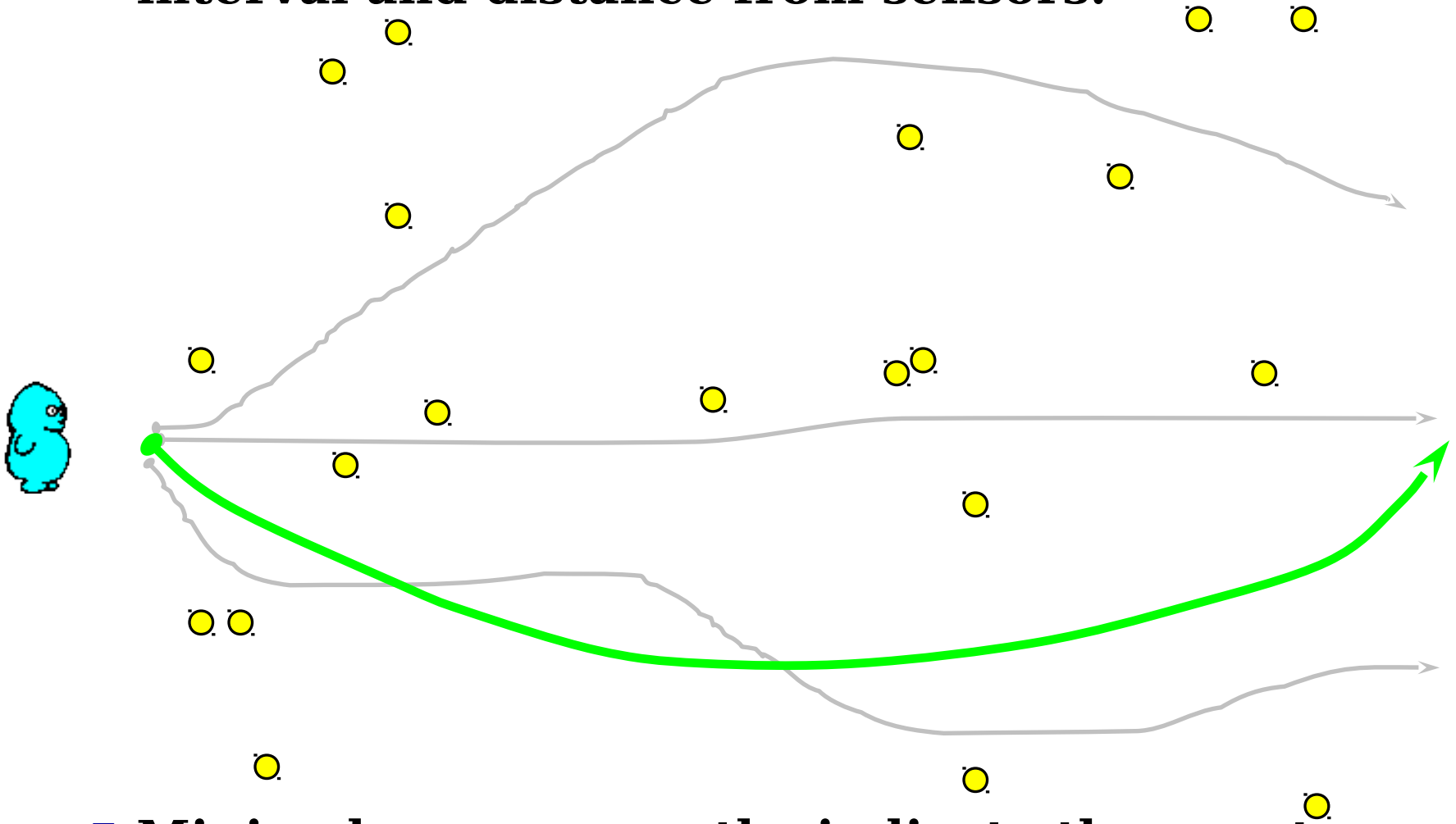
E.g.: Max Breach Path in a 50-



Virginia Tech's GUI

Exposure Model of Sensors

- **Likelihood of detection by sensors is a function of time interval and distance from sensors.**



- **Minimal exposure paths indicate the worst case scenarios in a field:**
  - **Can be used as a metric for coverage**
    - **Sensor detection coverage**
    - **Also, for wireless (RF) transmission coverage**

Exposure Model of Sensors (contd.)

- **Sensing model S at an arbitrary point P for a sensor s**

$$S(s,p) = \frac{\lambda}{\left[d(s,p)\right]^{K}}$$

where d(s,p) is the Euclidean distance between the sensor s and the point p, and positive constants λ and K are technology- and environment-dependent parameters.

- **Effective sensing intensity at point *p* in the sensor field F**

$$I_A(F,p) = \sum_{1}^{n} S(s_i,p)$$

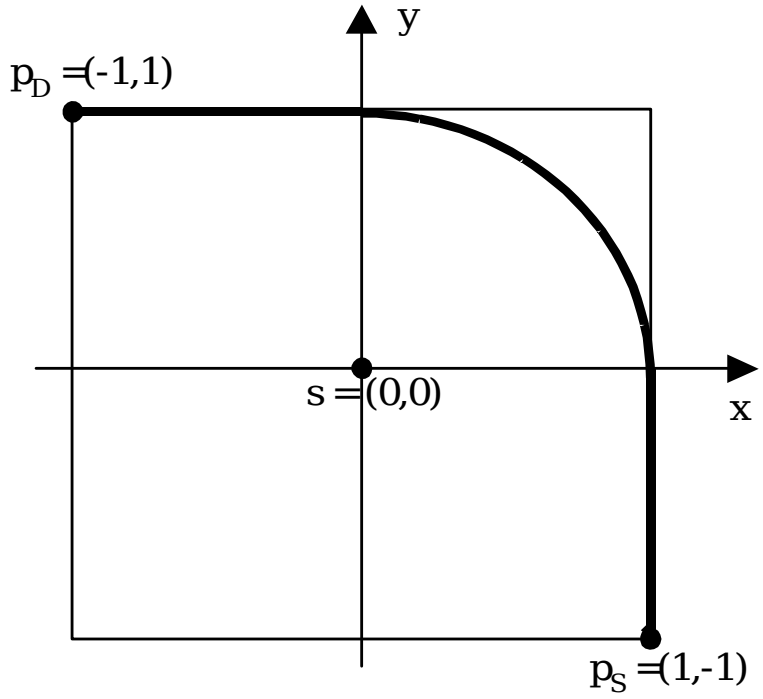
  - **All sensors**
  - **Closest sensor**
  - **K closest sensor**

- **The *Exposure* for an object *O* in the sensor field during the interval [t_1,t_2] along the path *p(t)* is**

$$E(p(t), t_1, t_2) \triangleq \int_{t_1}^{t_2} I(F, p(t)) \left| \frac{dp(t)}{dt} \right| dt$$

Minimum Exposure Path Formulation

- **Problem: find the path between two given points along which the exposure is smallest**
- **Example: minimum exposure for one sensor in a square field**

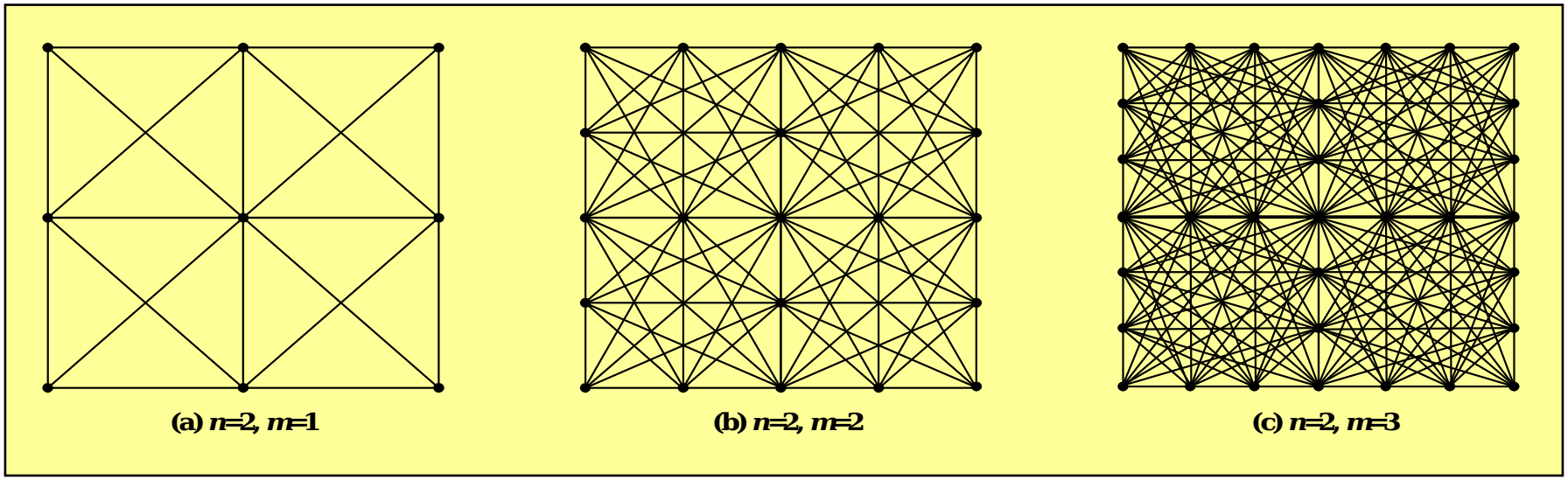


$$S(s(0,0), p(x,y)) = \frac{1}{d(s,p)} = \frac{1}{\sqrt{x^2 + y^2}}$$

$$E = \int_0^1 \frac{1}{\sqrt{x(t)^2 + y(t)^2}} \sqrt{\left[\frac{dx(t)}{dt}\right]^2 + \left[\frac{dy(t)}{dt}\right]^2}\, dt$$

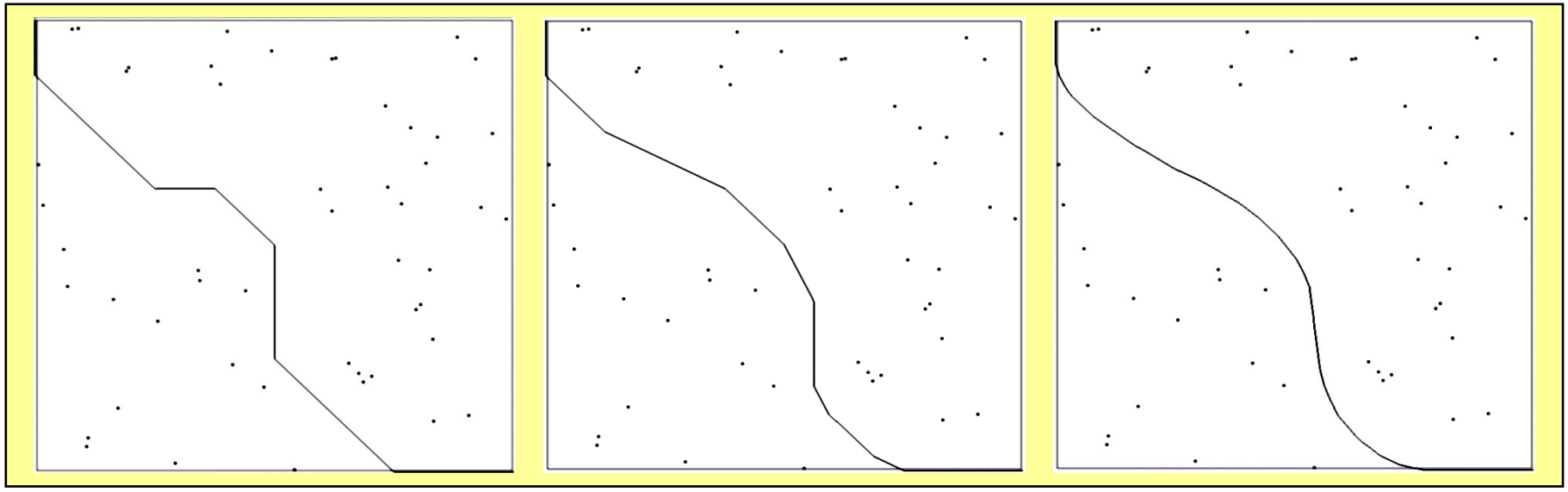
Solution Approach

- **General Case is analytically intractable**
- **Our approach: efficient and scalable method to approximate exposure integrals and search for Minimum Exposure paths**
  - **use a grid to approximate path exposures**
  - **exposure (weight) along each hrif edge approximated numerically**
  - **use Dijkstra's Single-Source Shortest Path Algorithm on the weighted graph (grid) to find the *Minimal Exposure Path***
  - **worst case search $O(n^2m)$ for a nxn grid with m divisions per edge**
    - **cost dominated by grid construction**
- **Generalized grids provide improved accuracy by increasing grid divisions at the cost of higher storage and run-time**



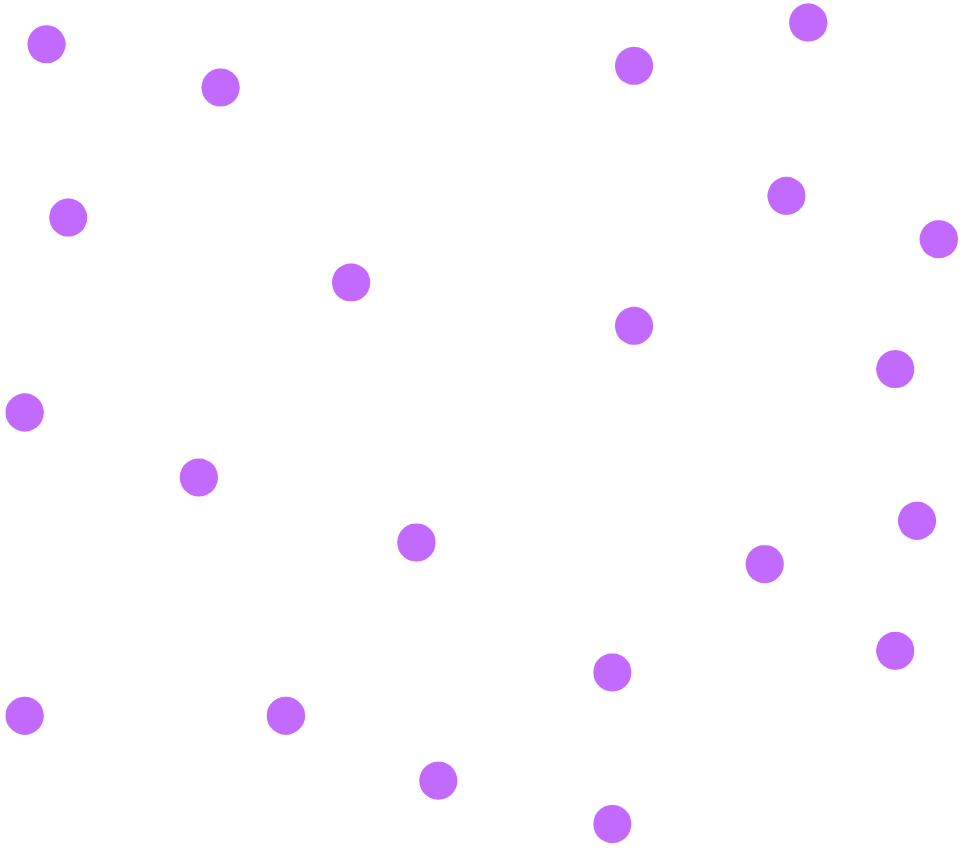
Status

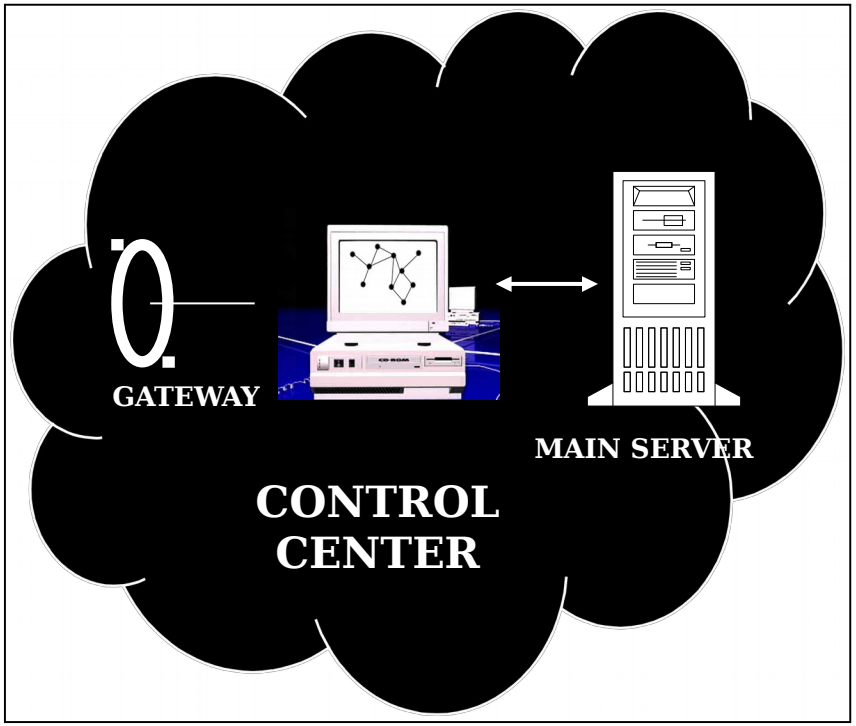
- **Centralized coverage server**
- **Integrated with the SensIT GUI (V. Tech.)**
  - **GUI passes node location, server reports back the desired path**
- **Part of the SITEX demonstration in Spring 2001**
- **Example: 50 randomly deployed node with the all-sensor intensity model**



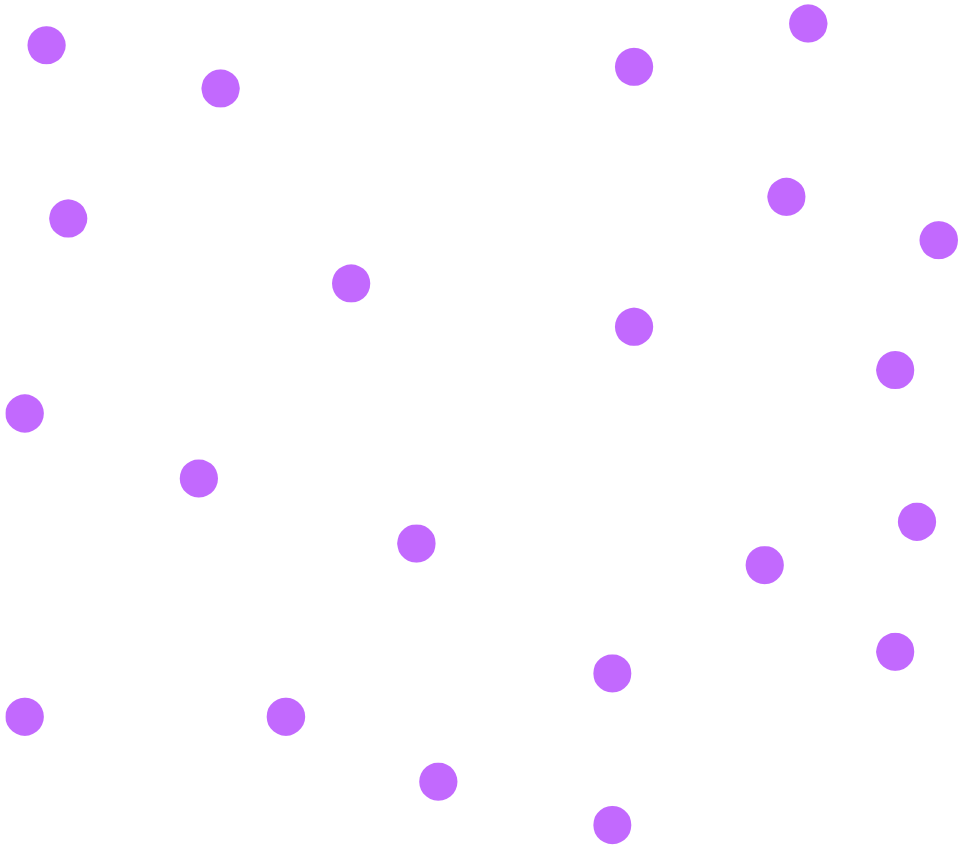
8x8 m=1	16x16 m=2	32x32 m=8
Exposure: 0.7079	Exposure: 0.6976	Exposure: 0.6945
Length: 1633.9	Length: 1607.7	Length: 1581.0

Problem? …. Centralized





Solution?



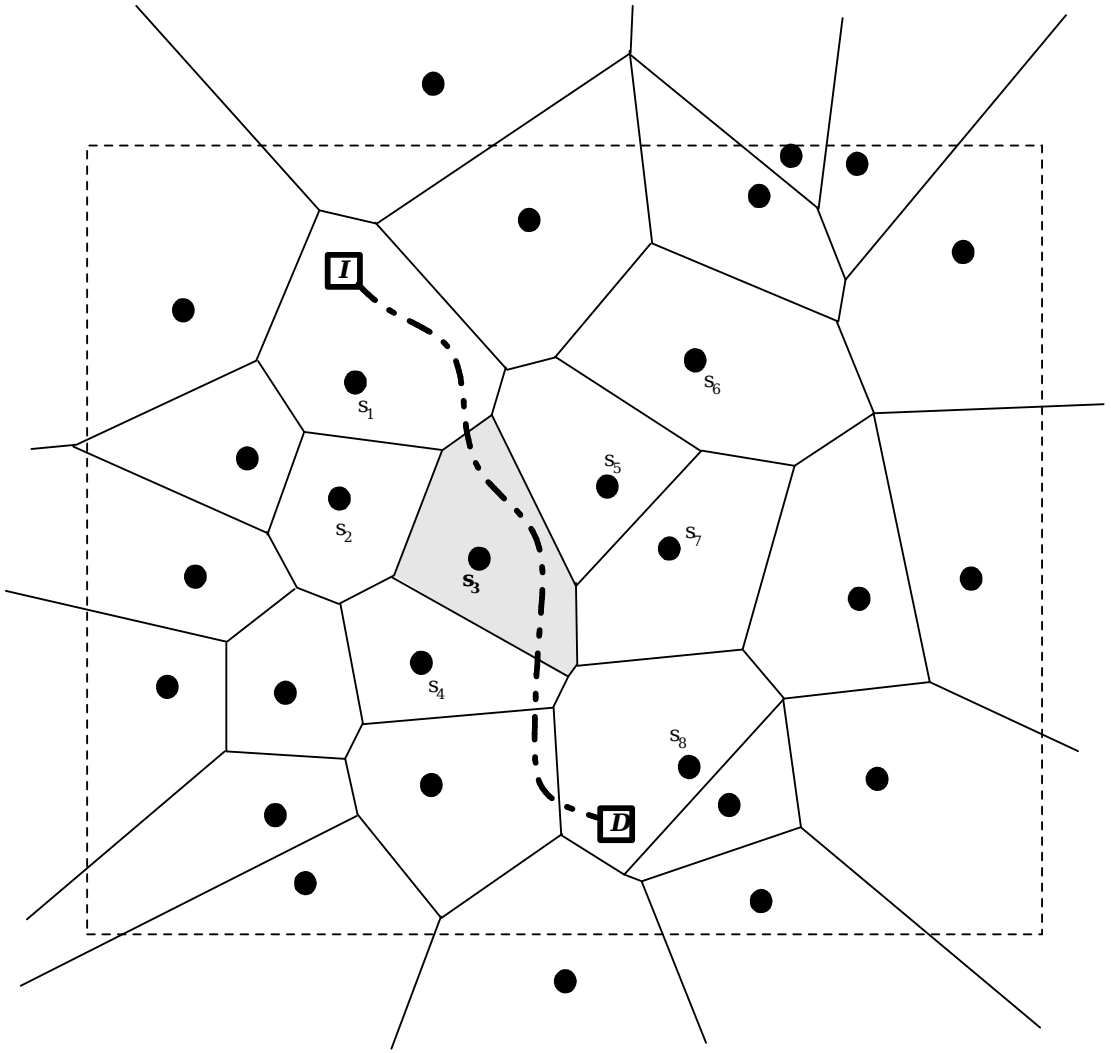
Localized Distributed Algorithm

Localized Algorithms

- Solve a distributed optimization problems
- Take into account topology, available energy, power etc.
- Obtain only needed information and use it to guide optimization
- Take into account problem properties
- Problems: Numerical errors

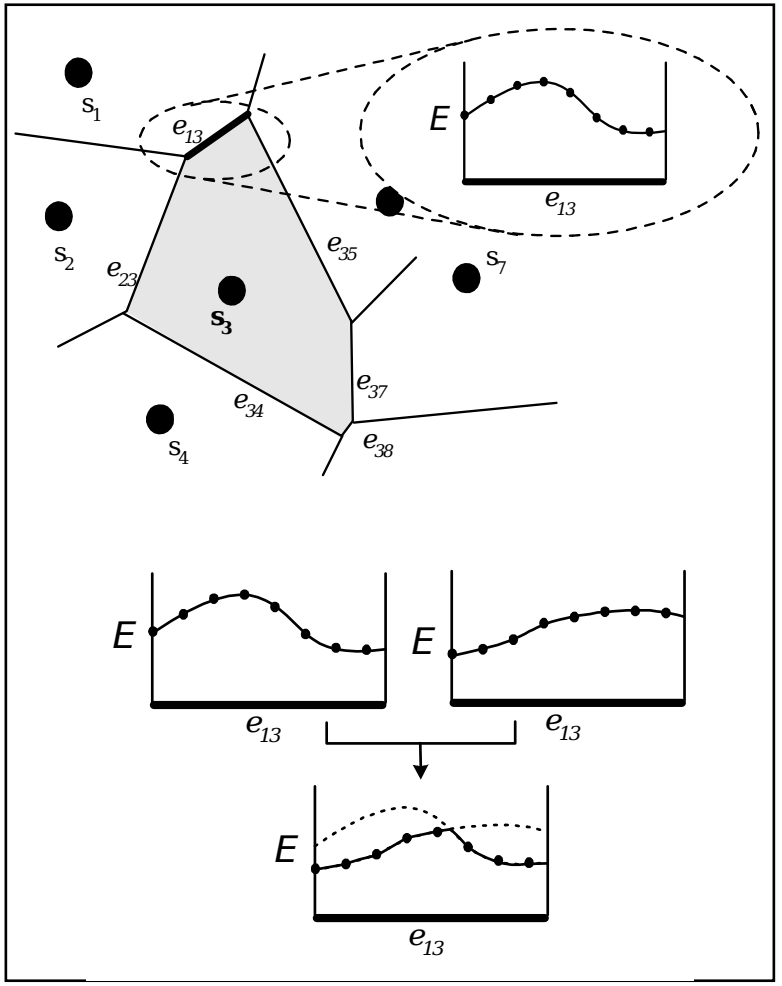
Localized Exposure

- **Voronoi Partitioning**
- **Advantages:**
  - One sensor per Polygon
  - Node can calculate its VP by knowing only its immediate (Delaunay) neighbors
  - Smaller VP's in high node density areas
- **Drawbacks**
  - One sensor potentially in charge of large area
  - Paths likely to be close to border edges
  - How to find Delaunay neighbors?
  - If node only knows locations of the Delaunay neighbors, then exposure calculation is not accurate



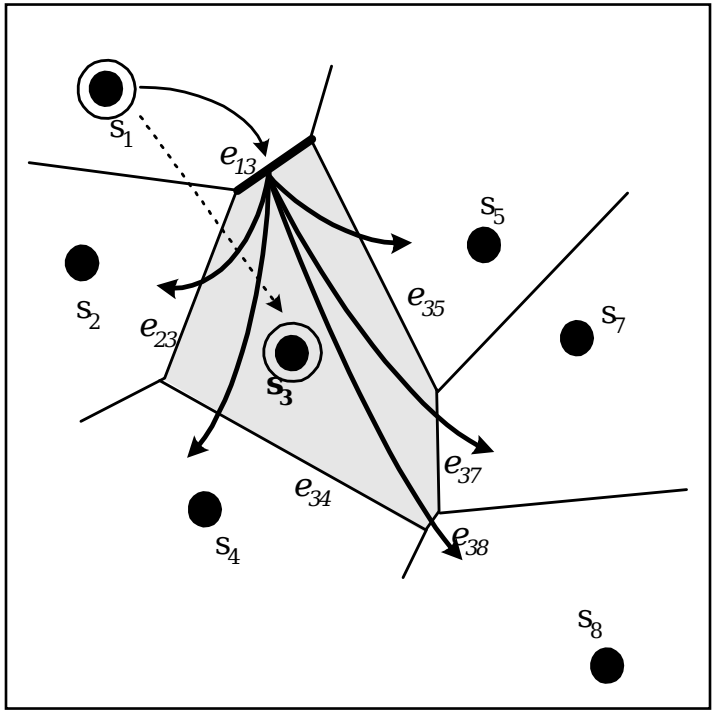
Localized Exposure (contd.)

- **Each polygon edge has a corresponding** *Exposure Profile (EP)*
- **Can use different data structures to store EPs.**
- **EPs initialized to infinity**
- **Continuously updated in algorithm by keeping smaller values and discarding larger ones**



Localized Exposure (contd.)

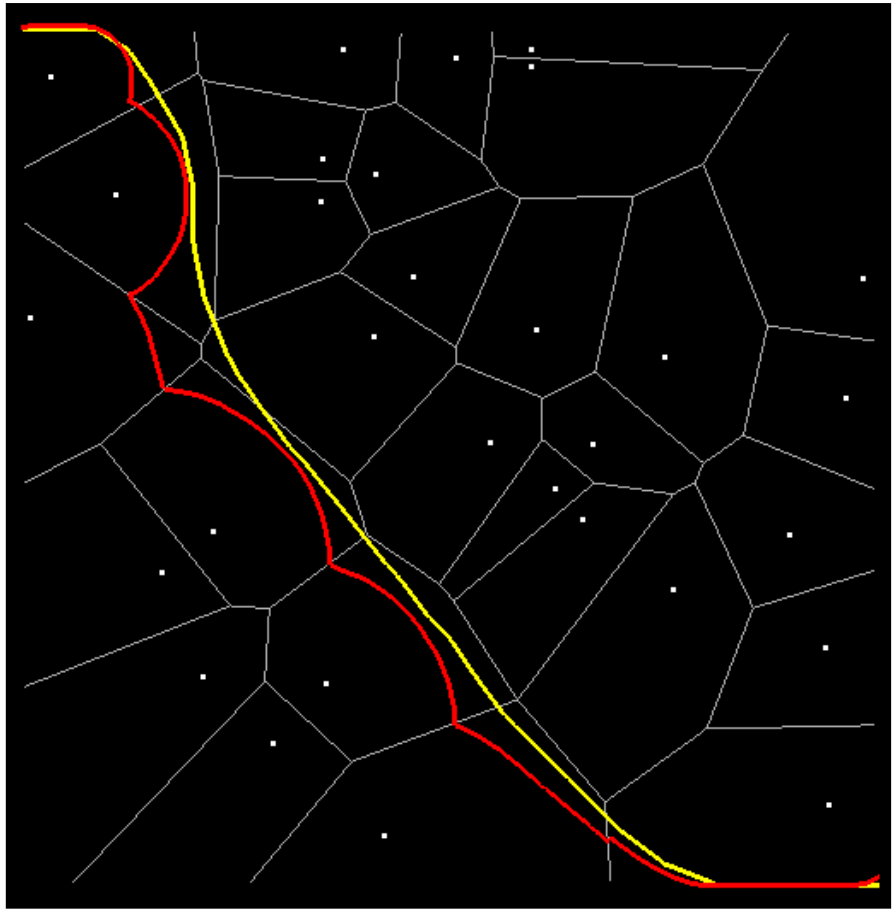
- **Node *s1* updates an EP *e13***
- ***s1* sends update message to neighbor node *s3***
- ***s3* computes new minimal exposure paths and updates all its EPs.**
- ***s3* sends appropriate EP update messages to corresponding neighbors**

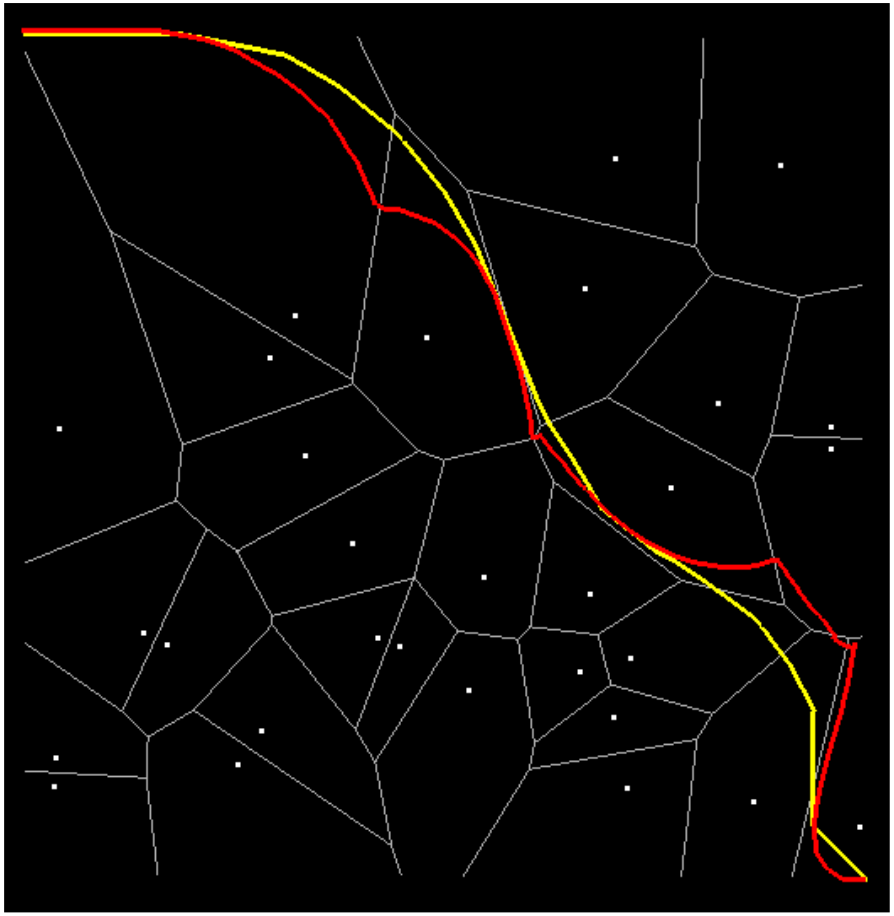


Localized Exposure (contd.)

- **Algorithm stops when**
  - **Each EP at the search boundary is larger than the specified termination condition (parameter indicating bound on exposure)**
    - **Specified by the algorithm at first**
    - **Periodically set to exposure at destination point during the optimization process (broadcast)**
  - **No more edge updates (EP)**
  - **Guaranteed to converge since exposure is always increasing.**
- **Message types**
  - Path_request**: Node s_i receives a request from an agent to find P_{minE} from I to D .**
  - Edge_update**: Node s_i receives an update notification from a neighbor to continue search for $P_{minE}(I,D)$.**
  - Abort_update**: Aborting conditions notification.**
  - Dest_update**: Destination reached notification**

Some Simulation Results

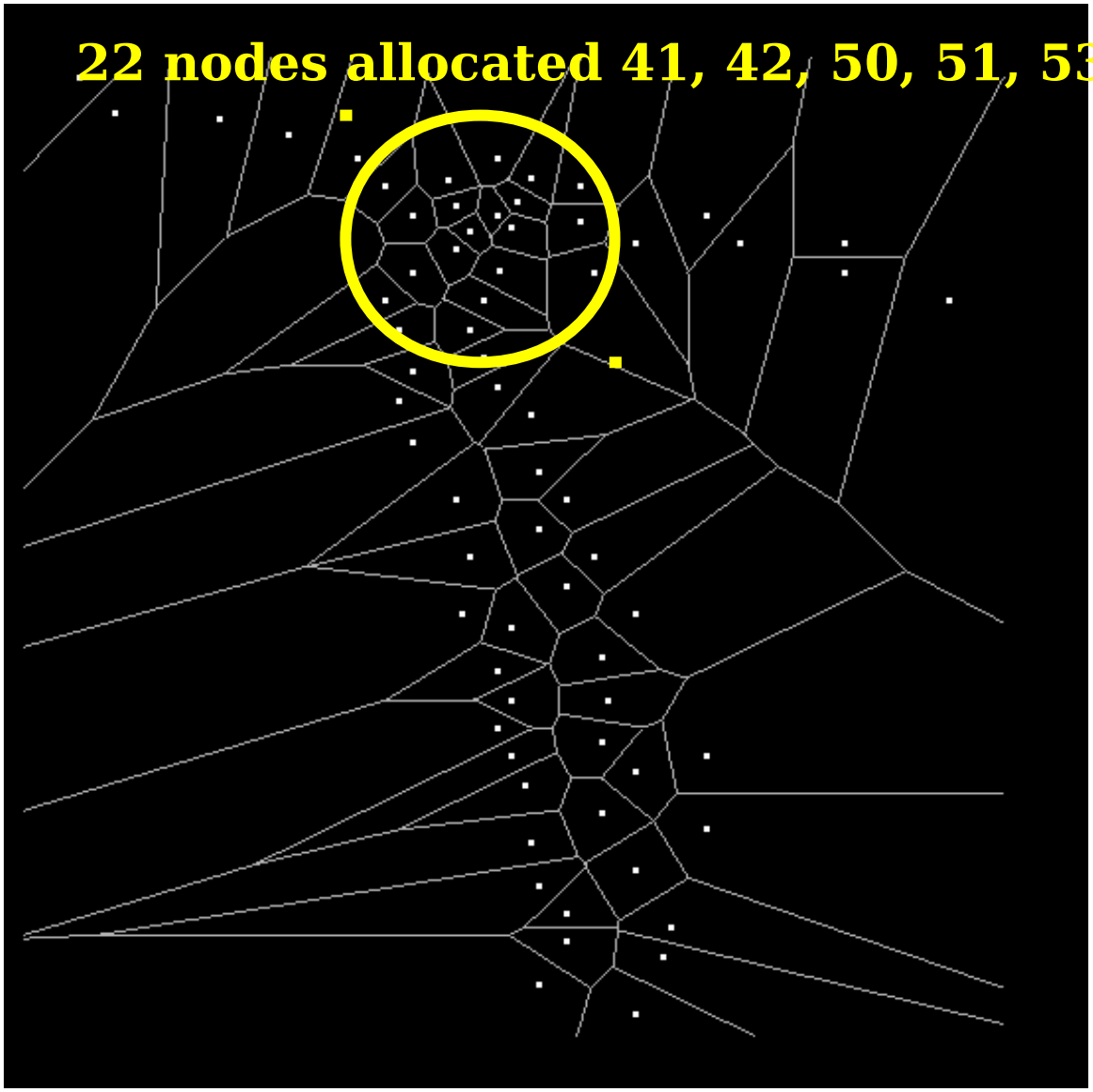




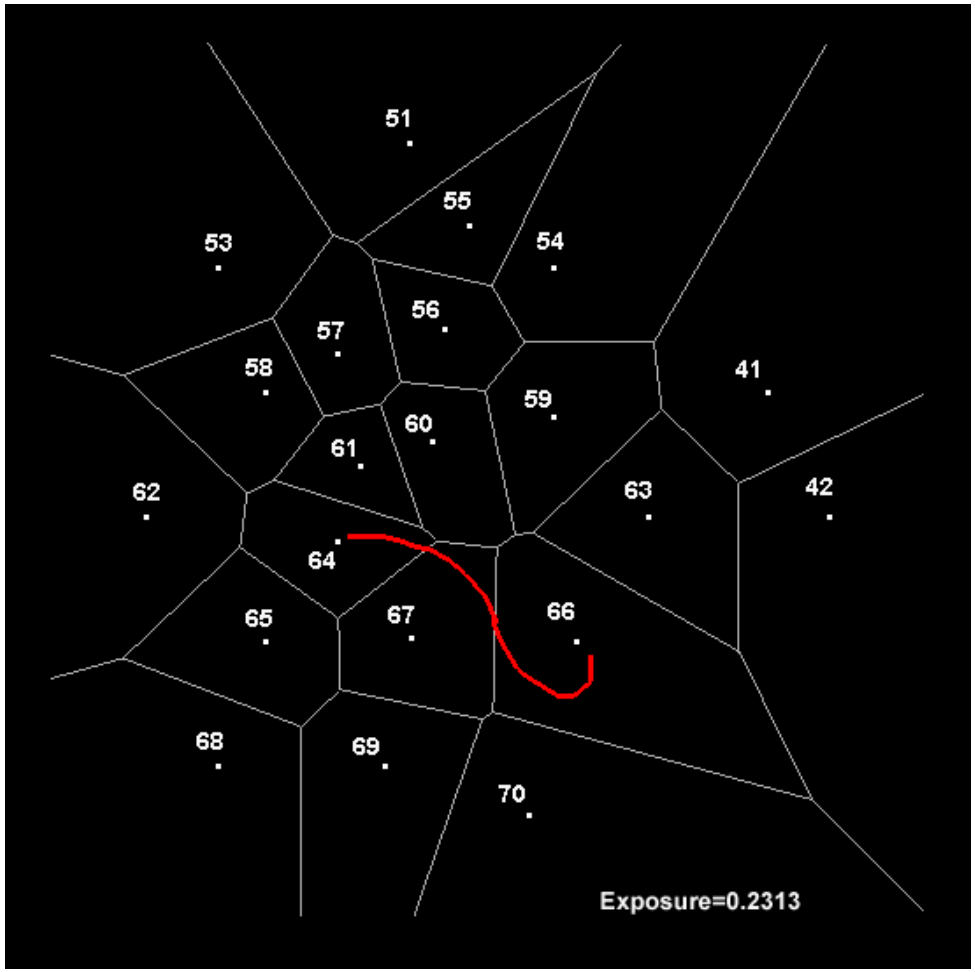
Status

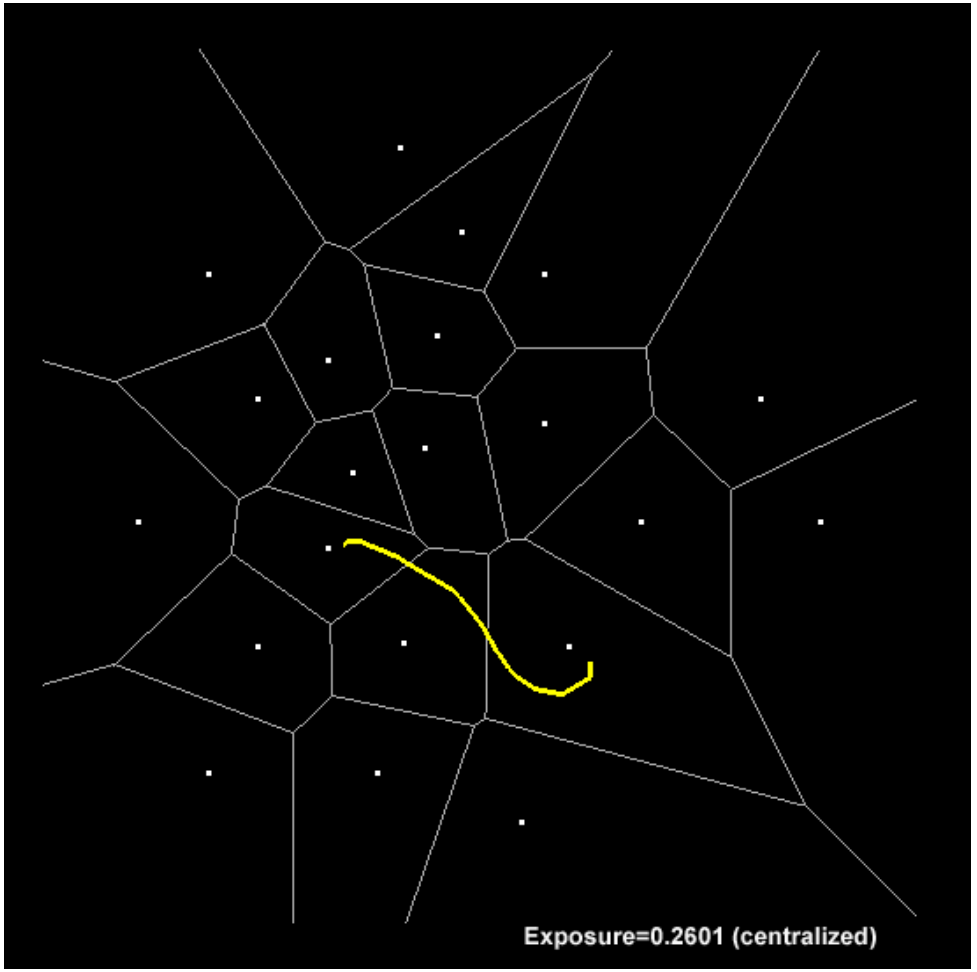
- **Initial implementation on Sensoria's WINS nodes**
  - **"Coverage Server" at each node**
  - **Listens for user query**
    - **request for minimum exposure path**
  - **Participates in distributed computation**
  - **Limitations/issues**
    - **one query at a time**
    - **uses an id-based addressing/routing emulated on top of diffusion**
- **Conducted experiments at SITEX demo on November 12, 2001**
  - **largest experiment: cluster off 22 nodes allocated 41, 42, 50, 51, 53-70**
  - **worked, but radio hanging problems on the nodes forced using the control ethernet for inter-node communication**

Results from SITEX Experiments



Results from SITEX Experiments

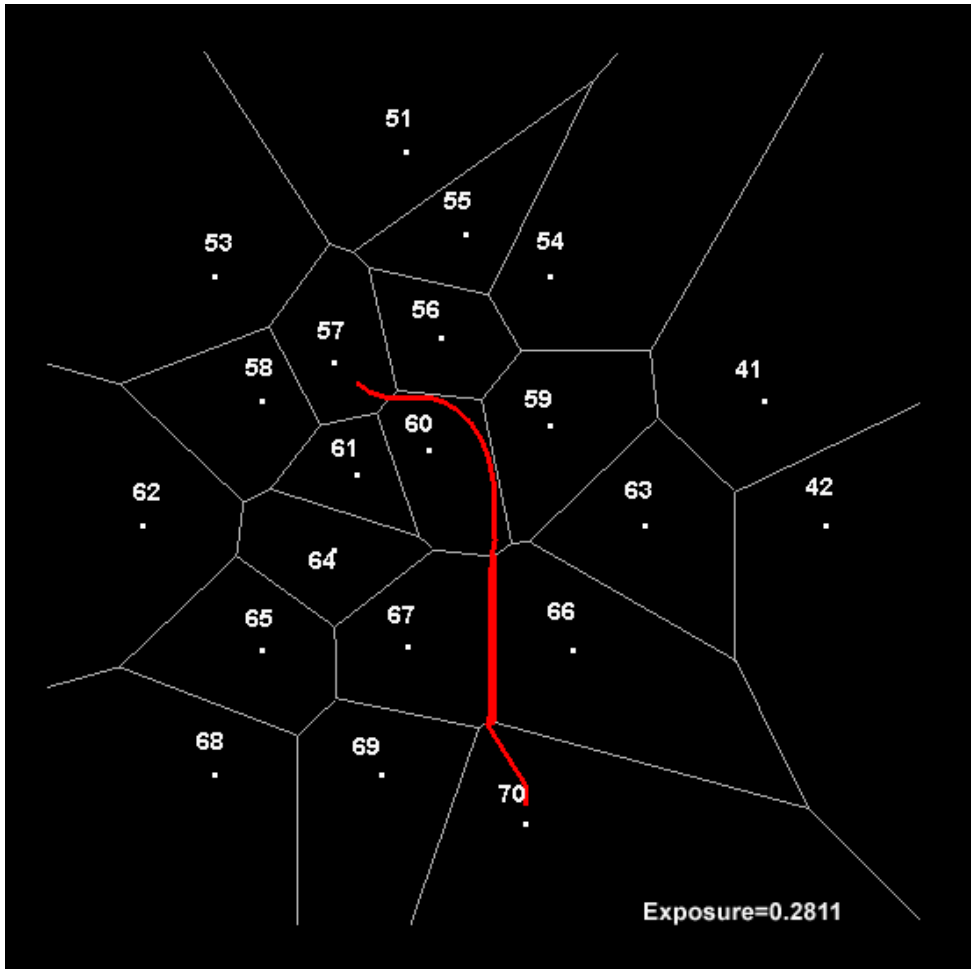


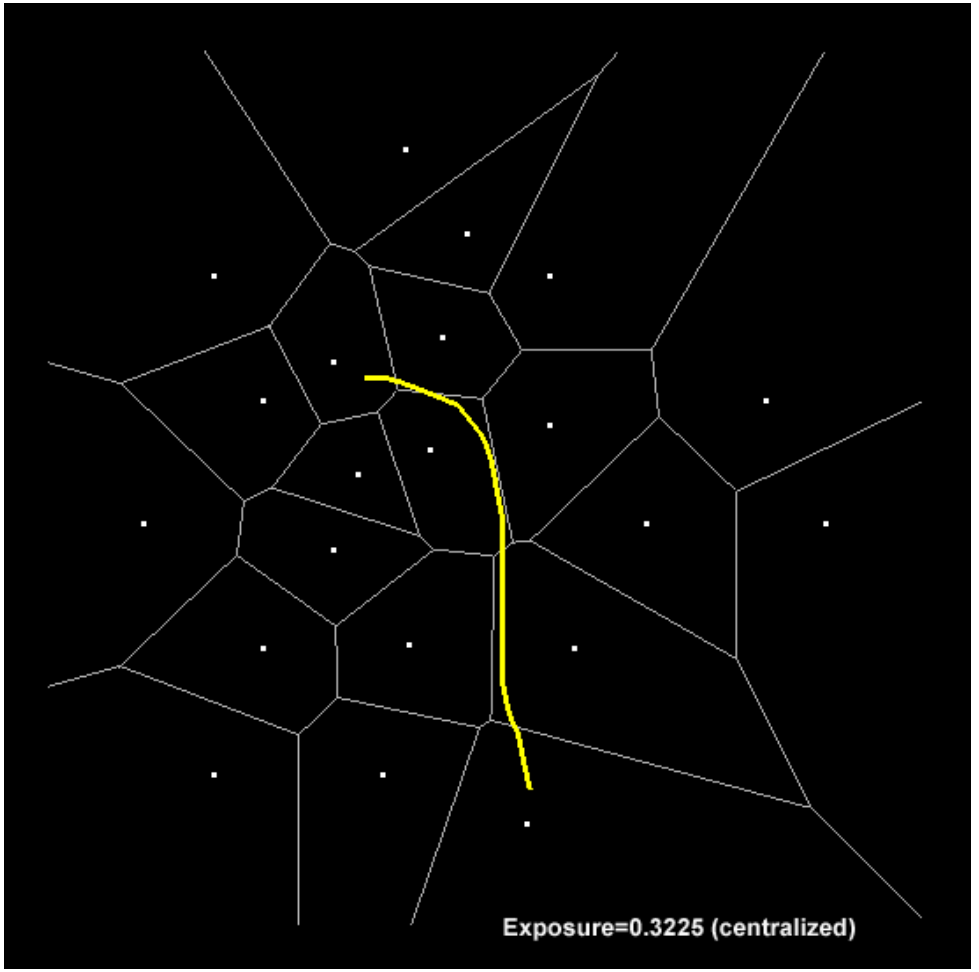


alized Implementation

Optimum (Simulated

Results from SITEX Experiments

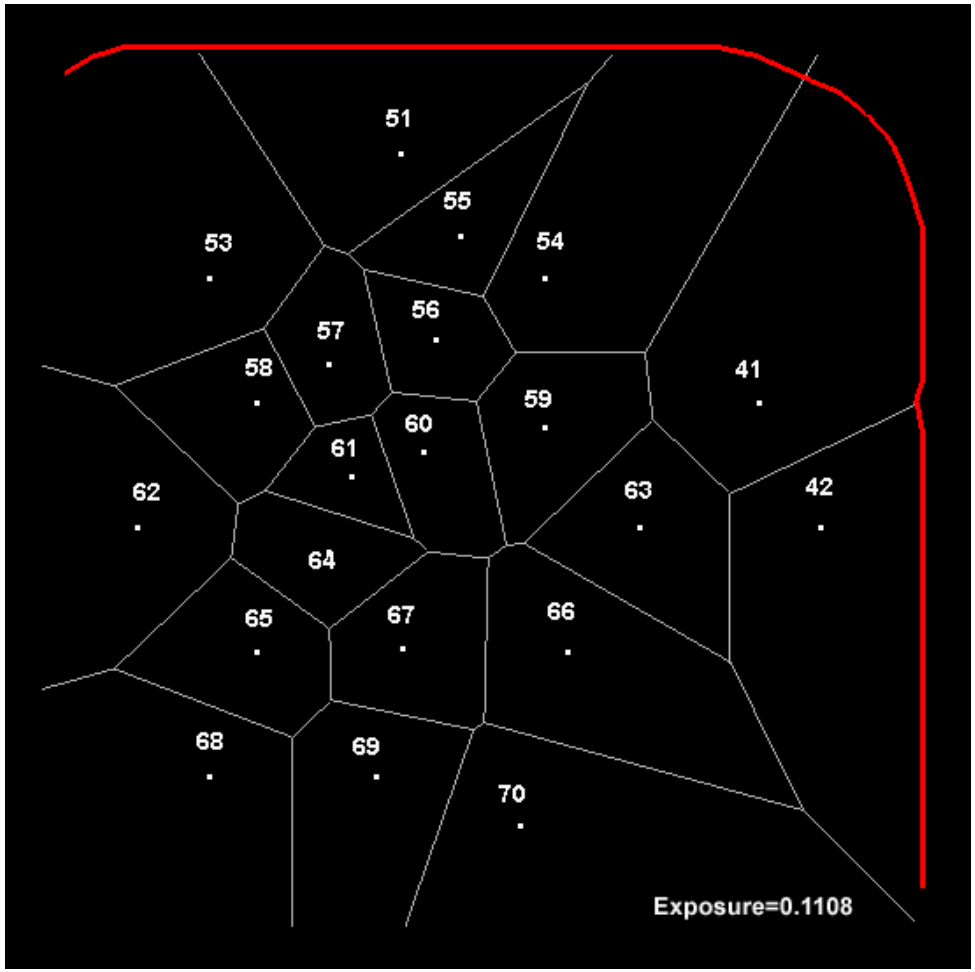


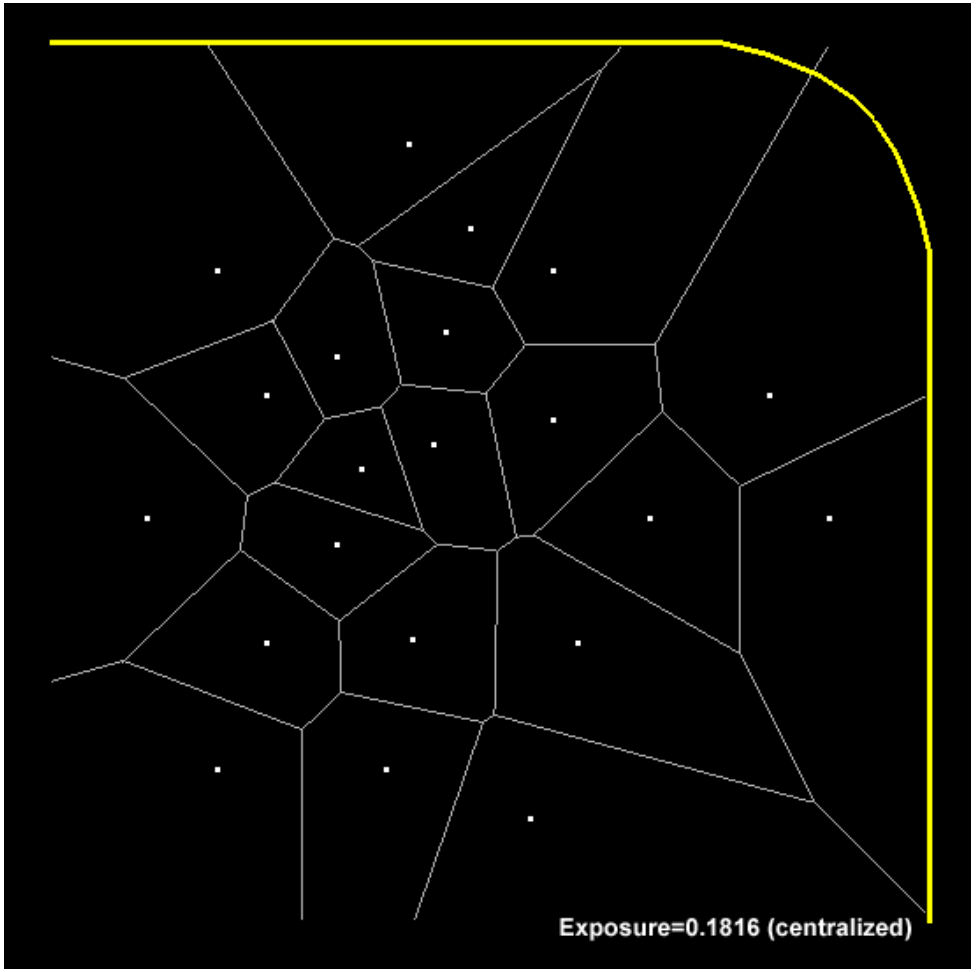


alized Implementation

Optimum (Simulated

Results from SITEX Experiments





alized Implementation

Optimum (Simulated